Research to Operations of Ionospheric Scintillation Detection and Forecasting

James C. Jones
Northrop Grumman

Kevin Scro
Air Force Space Command, Space and Missile Systems Center,
Remote Sensing Systems Directorate

David R. Payne, Ryan Ruhge, Dr. Brandon Erickson, Sage Andorka,
Jody Karmann, and Drew Ebelhar
Northrop Grumman

Nelson A. Bonito, Ronald G. Caton, Dr. Michael J. Starks
Air Force Research Laboratory, Space Vehicles Directorate, Battlespace Environment Division

CONFERENCE PAPER

Ionospheric Scintillation refers to random fluctuations in phase and amplitude of electromagnetic waves caused by a rapidly varying refractive index due to turbulent features in the ionosphere. Scintillation of trans-ionospheric VHF/UHF satellite communications and L-Band navigation radio frequency signals is particularly troublesome. This phenomenon degrades signal strength and integrity, negatively affecting satellite communications and navigation, radar, or radio signals from other systems that traverse or interact with the ionosphere. Although ionospheric scintillation occurs in both the equatorial and polar regions of the Earth, the focus of this modeling effort is equatorial scintillation. The ionospheric scintillation model is data-driven in the sense that scintillation observations are used to perform detection and characterization of scintillation structures. These structures are then propagated to future times using drift and decay models to represent the natural evolution of ionospheric scintillation. The impact on radio signals is also determined by the model and represented in graphical format to the user. A frequency-scaling algorithm allows for impact analysis on frequencies other than the observation frequencies. The project began with lab-grade software, and, through a tailored Agile development process, deployed operational-grade code to a DoD operational center. The Agile development process promotes adaptive planning, evolutionary development, early delivery, continuous improvement, regular collaboration with the customer, and encourages rapid and flexible response to customer-driven changes. The Agile philosophy values individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a rigid plan. The result was an operational capability that met customer expectations. Details of the model and the process of operational integration are discussed as well as lessons learned to improve performance on future projects.

1. INTRODUCTION

Scintillation is a random modulation imparted to propagating wave fields by structures in the propagating medium [1]. More specifically, ionospheric scintillation refers to random fluctuations in phase and amplitude of electromagnetic waves caused by a rapidly varying refractive index due to turbulent features in the ionosphere. It is primarily influenced by the ionospheric nighttime F-layer at an altitude of 300–350 km and occurs at scale lengths from 5 meters to 10 kilometers [2]. Scintillation occurs only at night and is generally limited to a region from the magnetic equator to about 20 degrees to the North and South [3]. The disruptions are greatest at lower VHF/UHF frequencies (~200-400 MHz) and decrease in severity as frequency increases into the upper part of this range (1 GHz and above). However, during extreme solar activity such as solar maximum, the disruptions are significant at GPS frequencies (L1 at ~1.5 GHz). The disturbed signals from some or all satellites results in less precise geolocation and under certain conditions complete loss of GPS navigation capability can result from scintillation due to loss of lock on sufficient number of satellites [3]. Scintillation of trans-ionospheric VHF/UHF satellite communication (SATCOM) and L-Band navigation radio frequency signals is particularly troublesome since this phenomenon can negatively affect satellite communications, precision navigation and timing, and radar signals that

traverse or interact with the ionosphere. Amplitude scintillations induce signal fading, which can produce message errors in satellite communication systems and may cause the loss of position fix or degradation of accuracy in GPS navigation systems [4].

Detection, analysis, and forecasts of scintillation are needed to be certain that any communication outages are not caused by system failures or jamming. Forecasts of scintillation can be used to plan for alternate methods of communication should conditions warrant, or to plan military operations when the opposition may have issues with communication. In the 2000s, operational scintillation analysis (such as the L-band Scintillation product [5]) consisted of a graphical depiction of raw observational data with no physical model to interpret the results. Forecasts of scintillating regions were merely climatological probabilities of scintillation occurrence (UHF SATCOM application [5]). The drawback of these approaches was that there was no connection between the observed data and the forecast. This approach did not contain a set of equations to describe the physical processes of scintillation. The next advance came in the form of a data-driven model that used scintillation measurements from ground-based sensors monitoring space-based beacons called Operational Space Environment Network Display (OpSEND) [6]. This improvement married an observation with a physical model, presenting a concise depiction of observed scintillating regions. However, some deficiencies were noted in OpSEND which included: a failure to process observations under certain conditions for large time periods, inaccurate detection and geolocation of scintillation structures, and incorrect and discontinuous mathematical processes that prevented proper advection of scintillating regions for forecasts. Because of these problems, improvements were desired and fortunately, there were multiple research and laboratory grade models whose baselines were advanced enough for consideration in an upgraded operational model.

In 2013, the Air Force Research Laboratory (AFRL) with funding from the Air Force Space and Missile Systems Command (SMC), embarked on a project to determine the “best of breed” of each of the component parts of these laboratory models. The components of each model were evaluated for scientific accuracy and validated against measurements to determine the best method for its function in a scintillation model. The best performing components were used in a final design of a state-of-the-art scintillation model. The result was the Scintillation Nowcast and Forecast Technology (SNFT) baseline. This baseline was validated by AFRL and when compared to the OpSEND product demonstrated a 90% improvement in specifying scintillation [7]. SNFT was recommended for transition to operations at the Air Force Space Weather Operations Center.

2. SCINTILLATION NOWCAST AND FORECAST TECHNOLOGY BASELINE

The Scintillation Nowcast and Forecast Technology (SNFT) baseline is a data-driven equatorial scintillation model developed by AFRL [7]. SNFT is data-driven in the sense that scintillation observations are used to perform detection and characterization of scintillation structures. These structures are then propagated to future times using drift and decay models to represent the natural evolution of ionospheric scintillation. The impact on radio signals is also determined by the model and represented in graphical format to the user. A frequency-scaling algorithm allows for impact analysis on frequencies other than the observation frequencies. The output of the model is available in a geolocatable format for any system capable of web mapping services. A more detailed description of the components of the model follows.

Identification of Scintillating Structures

SNFT begins by ingesting scintillation measurements from two networks: Scintillation Network and Decision Aid (SCINDA) and Ionospheric Scintillation and Total electron content Observer (ISTO). The instrumentation in these ground-based networks consists of antenna/receiver combinations tuned to ultra-high frequency (UHF) and Global Positioning System (GPS) beacons. The variations in the signal amplitude are used to evaluate the presence of scintillation. The observations are stored in a database to maintain a short-term history for baseline calculations. At run time, the observation data are extracted for a 12-hour period. A data validator ensures all observations meet quality specifications. The data are then grouped by station and a baseline of signal variation is determined. These baseline values establish the noise level for each ground station reducing the chance of a noisy instrument contaminating the model results. The data are then filtered to remove any spurious observations. Scintillation structures are then identified based on a set of rules applied to find elevated signal variation values.

Propagation of Structures

Identified structures are stretched along magnetic field lines to fill all magnetic latitudes +/- 20 degrees from the magnetic equator along the field line. During this stretching process, the scintillation amplitude values are converted to electron density values and scaled according to the climatological values from the Wideband Model (WBMOD) [8] scintillation climatology model. The widths of structures are also determined from the observations. The fully-developed structures are then propagated in an eastward direction according to the climatological Richmond drift model [9] .

Decay Mechanism
Decay of scintillating regions is governed by an exponential equation. The decay rate is a function of propagation history and frequency. Decay only begins after 1:30 AM local time. After this time, the region decays in intensity in 15 minute intervals until the intensity drops below a predetermined threshold.

Display
Once the model determines location and properties of the scintillation structures, the model outputs all the geolocation information and impact evaluation results for display. Currently, the data are sent to the United States Air Force (USAF) Air Combat Command (ACC) 557th Weather Wing for display on their web interface, AFW-WEBS, as a classified product. The impacts are overlaid onto a map of the Earth, using three colors indicating whether low (green), moderate (yellow), or high (red) impacts should be expected from scintillation when trying to communicate to the ground from a certain satellites. These satellites are specified by a configuration file, which can be updated by the model operator when needed.

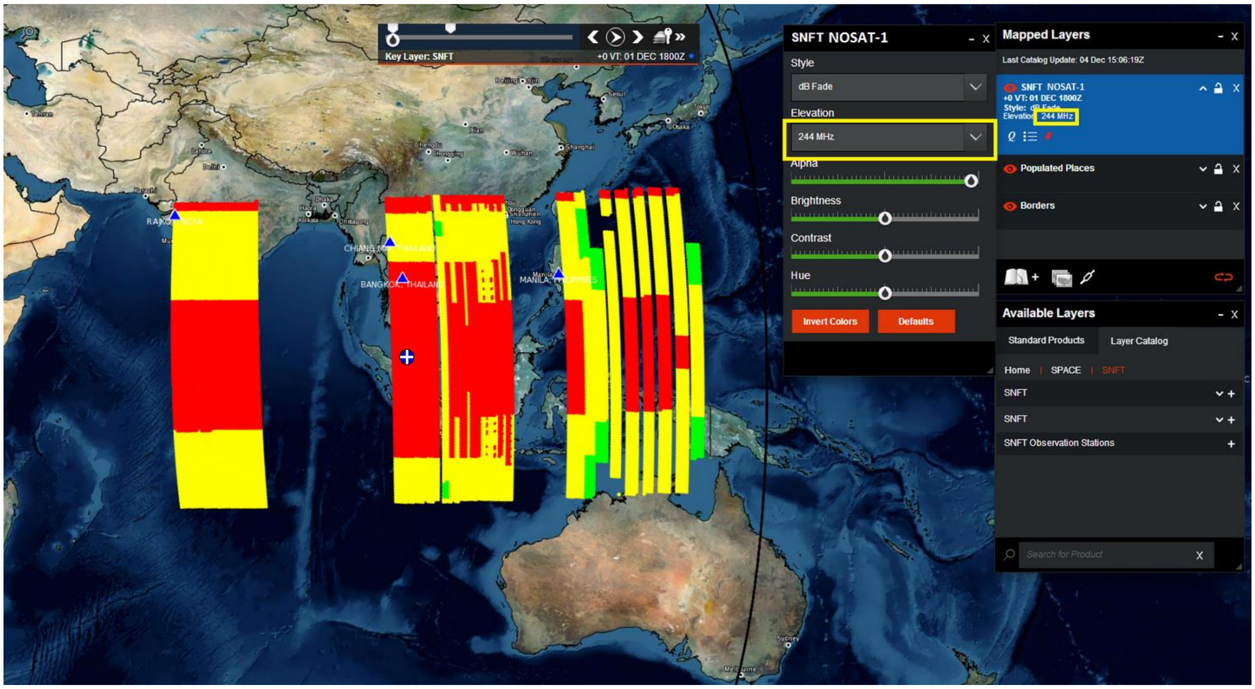

Figure 1. Notional SNFT Output

3. TRANSITION METHODOLOGY

The National Space Weather Program (NSWP) Strategic Plan [10], issued by the Executive Branch's Office of the Federal Coordinator for Meteorology(OFCM), calls for "optimal customer service through the interaction among customers, space weather forecasters, and space weather research and development activities". It goes further by saying "an effective technology transition process is essential to bringing to bear the fruits of research and development on space weather forecasting". Transitioning a model from the laboratory to an operational environment requires significant change from the prototype's original design. Live data, which is less timely and sometimes filled with errors, can crash an unsuspecting laboratory model without proper checks. Sharing system resources with other operational models requires good behavior on behalf of the new model. Interaction with

standardized processes and other system resources requires detailed systems engineering knowledge unknown to the designers of the prototype. System security certifications and protections are often overlooked in the lab environment since the prototyping effort is generally about the science and not security. All these (and many other) concerns force wholesale changes in the prototype software. This change can be intimidating to the prototype developer since model validation may be impacted by changes. Managing "change" is one of the biggest hurdles in the research to operations transition. To address the NSWP committee's recommendation for an effective technology transition process, a methodology that manages this change is needed.

Agile software development methods allow requirements and solutions to evolve by collaboration between self-organizing, cross-functional teams. It promotes adaptive planning, evolutionary development, early delivery, continuous improvement, and encourages rapid and flexible response to change [11]. According to the Agile Alliance, the twelve principles of Agile [12] include:

- Our highest priority is to satisfy the customer through **early and continuous delivery** of a valuable system
- A **working system** is the primary measure of progress
- **Welcome changing requirements**, even late in development
- **Deliver** a working system **frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale
- Business people and developers must **work together** daily throughout the project
- Build projects around **motivated individuals**. Give them the environment and support they need, and trust them to get the job done
- The most efficient and effective method of conveying information to and within a development team is **face-to-face conversation**
- Agile processes **promote sustainable development**
- Continuous attention to **technical excellence** and good design enhances agility
- **Simplicity**--the art of maximizing the amount of work not done--is essential
- The best architectures, requirements, and designs emerge **from self-organizing teams**
- At regular intervals, the team reflects on how to **become more effective**, then tunes and adjusts its behavior accordingly

Agile methods are about managing the impact of change, which works quite well for transitioning a model from the laboratory to an operational center. There are always unforeseen circumstances during this transition which oftentimes lead to catastrophic results. For example, data feeding the model in the lab tends to be available consistently and of good quality. Contrast that with data collection in real time where it can be late, missing, or garbled. Data handling in these two environments should be treated in two very distinct ways. For a research to operations transition, the key is the ability to embrace change and minimize the negative impacts of that change. Managing change is possible by valuing:

- **Individuals and interaction** over process and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

Agile focuses on delivering working code by incremental development steps with short iterations. The number of completed features in the software, which is forced by practice to have an open and flexible design, measures progress of the effort. The team members are empowered to decide for themselves how best to approach the problems at hand. Personal communications are encouraged among team members as well as with the customer. Transparency is the key to success. This transparency produces trust and in the end produces a better product for the customer.

Prior to the start of a project, the Product Owner meets with the development team. The team includes the Product Owner (or customer), the Scrum Master, a project manager, developers, and other stakeholders. They develop a release plan that determines project expectations such as "what will be delivered", "how will the work be delivered", "how often will deliveries be made", and "what is the definition of done". The answers to these questions determine the key milestones of the project and contribute to the success of the project.

Agile methods break down the process into short cycles called Sprints. Each Sprint is loaded with tasks to add functionality. Sprints are generally two to four weeks in duration. For this transition, we chose to use three-week Sprints.

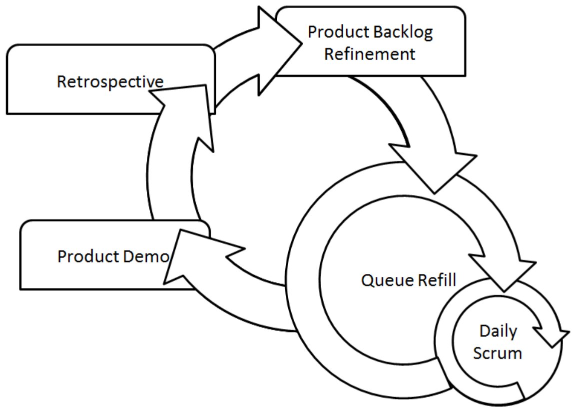

Figure 2. Sprint Cycle

At the beginning of each three-week Sprint Cycle (see figure 2), the team conducts the Product Backlog Refinement meeting. During this meeting, the team determines which tasks needed to be completed during the Sprint. Overall control of the task priority is determined by the Product Owner. This phase determines the direction of the project and sets the goals for the Sprint. This step is crucial for ensuring the customer gets what they want at the end of the project. Following the priorities set in the Product Backlog Refinement for the Sprint, a smaller group mainly consisting of the developers, the Project Manager, and the Scrum Master meets to determine the work for the next week in the Queue Refill meeting. These tasks come from the Product Backlog determined by the Product Owner. The developers, Project Manager, and Scrum Master gather each day for a Scrum. During this time, the team discusses what work was accomplished, what needs to be done, and any roadblocks preventing forward progress. The Scrum Master leads the discussion and takes action to remove any roadblocks preventing the team from moving forward. The developers then begin working the project tasks. Work continues in the cycle until near end of the Sprint and the Product Demo. In this demonstration of the new software release, the Product Owner and all stakeholders are shown the new features or changes developed during the Sprint. The Demo is a key component of Agile in that stakeholders provide feedback on the new features and potentially make recommendations for other features. Ideas generated during the Demo are captured and added to the Backlog for future consideration. Following the Product Demo, the Product Owner, Project Manager, developers, and Scrum Master complete a Retrospective to reflect on what went right/wrong during the last Sprint, identify key performers, and set the tone for the next Sprint. The Sprint Cycle is then complete but the project continues with the Product Backlog Refinement meeting to start the next Sprint. Work continues until the Product Owner and all Stakeholders are satisfied with the product.

The Agile development framework used by the SNFT project also featured an efficient web interface used to track the work that is done, to which all members of the team had access, enabling the Product Owner to check in and see how work is progressing at any time.

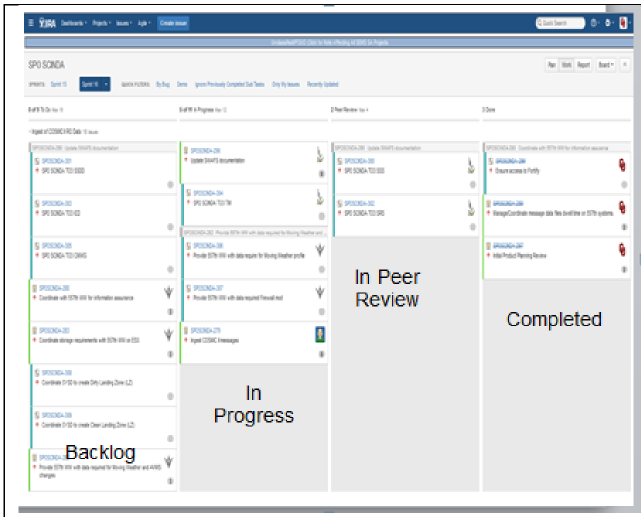

Figure 3. Rapid Board

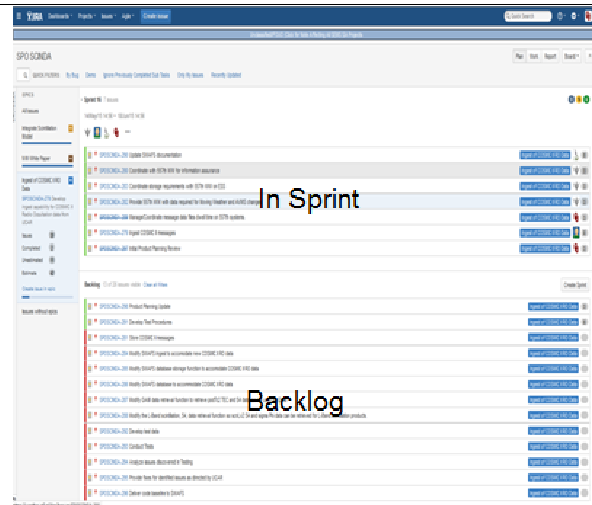

Figure 4. Planning Board

This interface is continually updated by the team and is used to ensure all tasks are being worked to meet the Product Owner's expectations. It features a Rapid Board (Fig. 3) which shows all the tasks and subtasks of the current sprint and their status (team backlog, work in progress, peer review, or completed) and a Planning Board (Fig. 4) which displays all tasks in the current Sprint as well as the remaining tasks in the Project Backlog. All new tasks identified by the team are discussed with the Product Owner during the Product Backlog Refinement meeting after being added to the planning board backlog. Developers work on tasks in the current Sprint and are able to divide these tasks into subtasks to more specifically describe what work needs to be done. The interface provides a means to track work in a simple and straightforward manner, enabling the developers to report the work being done in a short amount of time.

4. OPERATIONAL IMPLEMENTATION

The initial plan for the research to operations transition was to modify only the input data stream, output destination, and visualization methodology—preserving the as-delivered integrity of the science code within the model. However, as with any research to operations project, challenges were encountered along the way. The Agile process enabled the team to embrace the required changes, minimize the impact of change, and produce a better final product.

One of the first tasks was to perform a security analysis on the model's source code. These scans revealed numerous vulnerabilities (e.g. command line injection, external control of file system, etc.) that needed to be corrected. These vulnerabilities were prioritized by the Product Owner and other stakeholders and the team began working to correct the deficiencies. Additionally, code optimization software discovered numerous occasions of uninitialized variables that caused model instability. Configuration files describing the observation network, analysis regions, satellites, and ground stations that had been setup for laboratory testing differed significantly from the operational network and mission requirements. The impact of the misconfiguration caused the model to log numerous errors when connected to the operational data stream—enough errors to cause the log to fill the allocated disk space and crash the machine it was running on. This is an unacceptable feature in an operational setting. Other technical details of the model were found to be deficient during the verification phase of the integration effort. As with the security vulnerabilities, the team identified the issues, identified solutions, opened the tasks, and discussed possible solutions with the Product Owner and stakeholders (which included the original model developers). The Product Owner decided which tasks were important, prioritized accordingly and the team worked the tasks to produce an improved final product. The key to this success was embracing change and minimizing the impact of that change.

Using a traditional acquisition methodology with negotiated contracts, the tasks for an integration project would be clearly documented before the project started. The terms and conditions of the contract would be determined based

on the predicted task prioritization scheme. Many of the unforeseen challenges would have been out of scope in this situation and the project would halt or press forward with a flawed final product. Using Agile, the contract negotiation is much easier and allows the project to embrace unforeseen changes, incorporate the changes, and produce a superior final product.

5. LESSONS LEARNED

While the project was successful in delivering an improved capability to operations, there were still opportunities for improvement. The team documented lessons learned throughout the project and reviewed these lessons at completion. The most important lessons learn are described here.

Reach back to scientists is important. Entering into the project, the consensus was the model was ready for operations and should only require minimal modification for success. As unexpected issues arose, the Product Owner had to arrange for participation from the original model developers. This caused delays for some tasks until they were available for consultation. Afterwards, when access to these stakeholders was assured, minimal delays were encountered, and they contributed to some of the solutions and verified the integrator's solutions. The lesson learned here is that the original model developers should be a part of the integration team from start to finish.

The System Engineers and Software Developers that work on the targeted infrastructure need to be involved in model development much earlier than the transition. Generally, scientists develop models. These prototype models have a significant dollar value that includes the labor and validation expenses. The general perception is that any changes to the baseline would invalidate the prototype rendering it less capable. However, these models are generally not capable of functioning in an operational environment and often suffer from pitfalls of overlooking good software engineering design. If the engineers and developers that maintain the operational infrastructure are involved in the early design decisions, the transition to operations will be much easier. For one, less rework would be required during the transition. Security issues can be avoided with proper software design practices. Targeted infrastructure interfaces can be built into the prototype. If these practices are put into place, there would be no need to revalidate the model after the research to operation actions.

Considering these two lessons, we envision a design concept that includes representatives from both science and information technology (IT). Fig. 5 shows the relative contribution of scientists and IT engineers over the course of product development. The technical readiness level (TRL) is a systematic evaluation used to describe the maturity of technology [13]. The TRL ranges from (1) Basic Technology Research to (9) Fully Operational. See Table A1 in Appendix A for definitions of each TRL.

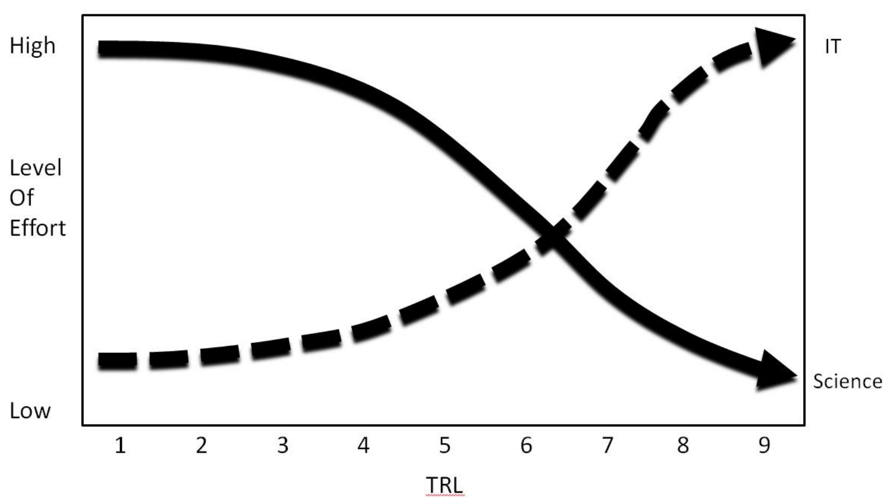

Figure 5. Expertise Participation vs TRL

At TRL 1, scientists perform the basic research to describe a physical process and begin to develop a prototype. From TRL 2–6, documentation of the scientific algorithms should be available (perhaps even in peer-reviewed

literature) to hand off to software engineers. This handoff should include algorithm description documents for key algorithms and any prototype code. The Software Engineers and Software Developers should be involved with some initial design decisions in the lower TRLs. At about TRL 6, a handover takes place where now the IT engineers take over the project to optimize the code for operational infrastructure. Their goal is to meet stringent requirements for operational performance (run time), security concerns, and integration into existing system components. The scientists should remain involved to ensure integrity of the science code and scrutinize all changes in this part of the code. Otherwise, the IT engineers continue with their design and development. This method should ensure a very robust baseline that contains the best algorithms the science can offer.

6. SUMMARY AND CONCLUSIONS

This project delivered much needed capability to space operators! The SNFT baseline identifies scintillation, spatial and temporally, 90% better than the previous operational model. It produces fewer false negative and false positive detection events and estimates scintillation structure size more accurately. Agile processes and tools were used during the technology transition from research to operations and provided the flexibility to deliver a superior product. Stakeholders remained involved throughout the process and the Product Owner made all decisions on the direction of the project and its priorities. The experience of having Engineers, Software Developers, and Scientists all supporting SNFT integration within a small team was exceptional allowing us to deliver a high quality product. Transparency was a fundamental tenant of the team. Demonstration of capability every three weeks, weekly queue refill meetings, and daily scrums ensured that transparency. Agreement up front on Definition of Done assured Developers and the Product Owner both understood the nature of the project. When issues arose, the team prioritized the tasks and stories to ensure requirements and the Definition of Done were met.

Special thanks to Jeff Cox of the Aerospace Corporation for his technical expertise during this project.

7. ABBREVIATIONS AND ACRONYMS

ACC - Air Combat Command
AFRL - Air Force Research Laboratory
AFW-WEBS - Air Force Weather Web Services
DoD - Department of Defense
GPS - Global Positioning System
ISTO - Ionospheric Scintillation and Total Electron Content Observer
NSWP - National Space Weather Program
OFCM - Office of the Federal Coordinator for Meteorology
OpSEND - Operational Space Environment Network Display
SATCOM - Satellite Communications
SCINDA - Scintillation Network and Decision Aid
SMC - Space and Missile Systems Command
SNFT - Scintillation Nowcast and Forecast Technology
UHF - Ultra High Frequency (300 – 3000 MHz)
VHF - Very High Frequency (30 – 300 MHz)
USAF - United States Air Force

8. REFERENCES

[1] C. L. Rino, The Theory of Scintillation with Applications in Remote Sensing, Hoboken, NJ: IEEE, 2011.

[2] A. D. Wheelon, Electromagnetic Scintillation, New York: Cambridge University Press, 2001.

[3] Boston College, "Ionospheric Modeling: Develpment, Verification, and Validation," Air Force Research Laboratory, Hanscom AFB, MA, 2007.

[4] Phillips Laboratory, "Integrated Efforts for Analysis of Geophysical Measurements and Models," Air Force Materiel Command, Hanscom AFB, MA, 1997.

[5] Air Force Weather, "Air Force Weather Space Weather Product Catelog," USAF, Omaha, Ne, 2015.

[6] G. Bishop, T. Bullett, K. Groves, S. Quigley, P. Doherty, E. Sexton, K. Scro, R. Wilkes and P. Citrone, "Operational Space Environment Network Display (OpSEND)," *Radio Science,* vol. 39, no. 1, 2004.

[7] Air Force Research Laboratory - Space Vehicles Directorate, "SPOSCINDA Analysis Results and Software Production," AFRL/RV, Kirtland AFB, NM, 2013.

[8] Northwest Research Associates, Inc., "An Improved Model of High-Latitude F-Region Scintillation (WBMOD v13)," Phillips Laboratory, Hanscom Air Force Base, MA, 1994.

[9] A. D. Richmond, M. Blanc, B. A. Emory, R. H. Wand, B. G. Fejer, R. F. Woodman, S. Ganguly, P. Anaenc, R. A. Behnke, C. Calderon and a. J. Evans, "An Emperical Model of Quiet-Day Ionospheric Electric Fields at Middle and Low Latitudes," *Journal of Geophysical Research,* vol. 85, no. A9, pp. 4658-4664, 1980.

[10] National Space Weather Program Council, "The National Space Weather Program Strategic Plan," Office of the Federal Coordinator for Meteorology, Washington, D.C., 2010.

[11] The Agile Alliance, 2015. [Online]. Available: http://www.agilealliance.org/. [Accessed 23 July 2015].

[12] The Agile Alliance, "Twelve Principles of Agile," The Agile Alliance, 2015. [Online]. Available: http://www.agilealliance.org/the-alliance/the-agile-manifesto/the-twelve-principles-of-agile-software/. [Accessed 28 July 2015].

[13] Assistant Secretary of Defense for Research and Engineering, "Technology Readiness Assessment Guidance," Department of Defense, Washington, DC, 2011 Revised 13 May 2013.

[14] K. Aguanno, Managing Agile Projects, Lakefield, Ontario, Canada: Multi-Media Publications Inc, 2004.

9. APPENDIX A

Table A1. Technical Readiness Level Definitions

TRL	Definition	Description	Supporting Information
1	Basic principles observed and reported.	Lowest level of technology readiness. Scientific research begins to be translated into applied research and development (R&D). Examples might include paper studies of a technology's basic properties.	Published research that identifies the principles that underlie this technology. References to who, where, when.
2	Technology concept and/or application formulated.	Invention begins. Once basic principles are observed, practical applications can be invented. Applications are speculative, and there may be no proof or detailed analysis to support the assumptions. Examples are limited to analytic studies.	Publications or other references that outline the application being considered and that provide analysis to support the concept.
3	Analytical and experimental critical function and/or characteristic proof of concept.	Active R&D is initiated. This includes analytical studies and laboratory studies to physically validate the analytical predictions of separate elements of the technology. Examples include components that are not yet integrated or representative.	Results of laboratory tests performed to measure parameters of interest and comparison to analytical predictions for critical subsystems. References to who, where, and when these tests and comparisons were performed.
4	Component and/or breadboard validation in a laboratory environment.	Basic technological components are integrated to establish that they will work together. This is relatively "low fidelity" compared with the eventual system. Examples include integration of "ad hoc" hardware in the laboratory.	System concepts that have been considered and results from testing laboratory-scale breadboard(s). References to who did this work and when. Provide an estimate of how breadboard hardware and test results differ from the expected system goals.
5	Component and/or breadboard validation in a relevant environment.	Fidelity of breadboard technology increases significantly. The basic technological components are integrated with reasonably realistic supporting elements so they can be tested in a simulated environment. Examples include "high-fidelity" laboratory integration of components.	Results from testing laboratory breadboard system are integrated with other supporting elements in a simulated operational environment. How does the "relevant environment" differ from the expected operational environment? How do the test results compare with expectations? What problems, if any, were encountered? Was the breadboard system refined to more nearly match the expected system goals?
6	System/subsystem model or prototype demonstration in a relevant environment.	Representative model or prototype system, which is well beyond that of TRL 5, is tested in a relevant environment. Represents a major step up in a technology's demonstrated readiness. Examples include testing a prototype in a high-fidelity laboratory environment or in a simulated operational environment.	Results from laboratory testing of a prototype system that is near the desired configuration in terms of performance, weight, and volume. How did the test environment differ from the operational environment? Who performed the tests? How did the test compare with expectations? What problems, if any, were encountered? What are/were the plans, options, or actions to resolve problems before moving to the next level?
7	System prototype demonstration in an operational environment.	Prototype near or at planned operational system. Represents a major step up from TRL 6 by requiring demonstration of an actual system prototype in an operational environment (e.g., in an aircraft, in a vehicle, or in space).	Results from testing a prototype system in an operational environment. Who performed the tests? How did the test compare with expectations? What problems, if any, were encountered? What are/were the plans, options, or actions to resolve problems before moving to the next level?
8	Actual system completed and qualified through test and demonstration.	Technology has been proven to work in its final form and under expected conditions. In almost all cases, this TRL represents the end of true system development. Examples include developmental test and evaluation (DT&E) of the system in its intended weapon system to determine if it meets design specifications.	Results of testing the system in its final configuration under the expected range of environmental conditions in which it will be expected to operate. Assessment of whether it will meet its operational requirements. What problems, if any, were encountered? What are/were the plans, options, or actions to resolve problems before finalizing the design?
9	Actual system proven through successful mission operations.	Actual application of the technology in its final form and under mission conditions, such as those encountered in operational test and evaluation (OT&E). Examples include using the system under operational mission conditions.	OT&E reports.